WSUS: czyszczenie i utrzymanie

Typ: fiszka **Dotyczy: Windows Server Update Services (WSUS)**

Efekt: odrzucamy zastąpione aktualizacje, czyścimy bazę WSUS z niepotrzebnych metadanych i plików, i ustawiamy to jako zadanie cykliczne zamiast robić to ręcznie za każdym razem.

Skróty użyte na tej stronie (4)

Skrót	Co znaczy
HTTP	Hypertext Transfer Protocol: protokół stron WWW
HTTPS	HTTP zabezpieczony szyfrowaniem TLS
SQL	Structured Query Language: język zapytań do relacyjnych baz danych
WSUS	Windows Server Update Services: firmowy serwer aktualizacji Windows

Fiszka

TYLKO ODCZYT

Uruchomienie skryptu odrzucającego zastąpione aktualizacje: Decline-Updates.ps1

```
.\Decline-Updates.ps1 -WSUSServer WSUSServer.Company.com -WSUSPort 8530
```

`WSUSServer.Company.com` to adres naszego serwera WSUS, `8530` to numer portu WSUS (domyślnie `8530` dla HTTP, `8531` dla HTTPS). Efekty działania: skrypt analizuje listę dostępnych aktualizacji, odrzuca te spełniające kryteria, a jeśli odrzucił cokolwiek — od razu uruchamia czyszczenie WSUS, żeby zwolnić miejsce.

WSUS = Windows Server Update Services: firmowy serwer aktualizacji Windows · **HTTP** = Hypertext Transfer Protocol: protokół stron WWW · **HTTPS** = HTTP zabezpieczony szyfrowaniem TLS

⚠ ZMIENIA SYSTEM

Treść skryptu

```
Param(
    [Parameter(Mandatory=$false,
    ValueFromPipeline=$true,
    ValueFromPipelineByPropertyName=$true,
    ValueFromRemainingArguments=$false,
    Position=0)]
    [string]$WSUSServer = "Localhost", #default to localhost
    [int]$WSUSPort=8530,
    [switch]$reportonly
    )

Function Decline-Updates{
    Param(
        [string]$WsusServer,
        [int]$WSUSPort,
        [switch]$ReportOnly
    )
    write-host "Connecting to WSUS Server $WSUSServer and getting list of updates"
    $Wsus = Get-WSUSserver -Name $WSUSServer -PortNumber $WSUSPort
    if($WSUS -eq $Null){
        write-error "unable to contact WSUSServer $WSUSServer"
    }else{
        $Updates = $wsus.GetUpdates()
        write-host "$(($Updates | where {$_.IsDeclined -eq $false} | measure).Count)
Updates before cleanup"
        $updatesToDecline = $updates | where {$_.IsDeclined -eq $false -and (
        $_.IsSuperseded -eq $true -or   #remove superseded updates
        $_.PublicationState -eq "Expired" -or #remove updates that have been pulled by
Microsoft
        $_.LegacyName -imatch "ia64" -or #remove updates for itanium computers (1/2)
        $_.LegacyName -imatch "x86" -or  #remove updates for 32-bit computers
        $_.LegacyName -match "XP" -or   #remove Windows XP updates (1/2)
        $_.producttitles -match "XP" -or #remove Windows XP updates (1/2)
        $_.Title -match "Itanium" -or   #remove updates for itanium computers (2/2)
        $_.Title -match "ARM64" -or   #remove updates for itanium computers (2/2)
```

```
        ($_.Title -imatch "language" -and $_.Title -notmatch "(PL|ENG)") -or  #remove
langauge packs
        $_.Title -match "Sprachpaket" -or
        $_.title -match "Internet Explorer 7" -or #remove updates for old versions of IE
        $_.title -match "Internet Explorer 8" -or
        $_.title -match "Internet Explorer 9" -or
        $_.title -match "Japanese" -or #some non-english updates are not filtered by
WSUS language filtering
        $_.title -match "Korean" -or
        $_.title -match "Taiwan" -or
        $_.Title -match "Beta" -or     #Beta products and beta updates
        $_.title -match "Embedded"     #Embedded version of Windows
        )}

        write-host "$(($updatesToDecline | measure).Count) Updates to decline"
        $changemade = $false
        if($reportonly){
            write-host "ReportOnly was set to true, so not making any changes"
        }else{
            $changemade = $true
            $updatesToDecline | %{$_.Decline()}
        }

        #Decline updates released more then 3 months prior to the release of an included
service pack
        # - service packs updates don't appear to contain the supersedance information.
        Foreach($SP in $($updates | where title -match "^Windows Server \d{4} .* Service
Pack \d")){
            if(($SP.ProductTitles |measure ).count -eq 1){
                $updatesToDecline = $updates | where {$_.IsDeclined -eq $false -and
$_.ProductTitles -contains $SP.ProductTitles -and $_.CreationDate -lt
$SP.CreationDate.Addmonths(-3)}
                if($updatesToDecline -ne $null){
                    write-host "$(($updatesToDecline | measure).Count) Updates to
decline (superseded by $($SP.Title))"
                    if(-not $reportonly){
                        $changemade = $true
                        $updatesToDecline | %{$_.Decline()}
                    }
```

```
                }
            }
        }

        #if changes were made, run a WSUS cleanup to recover disk space
        if($changemade -eq $true -and $reportonly -eq $false){
            $Updates = $wsus.GetUpdates()
            write-host "$(($Updates | where {$_.IsDeclined -eq $false} | measure).Count)
Updates remaining, running WSUS cleanup"
            Invoke-WsusServerCleanup -updateServer $WSUS -CleanupObsoleteComputers -
CleanupUnneededContentFiles -CleanupObsoleteUpdates -CompressUpdates -
DeclineExpiredUpdates -DeclineSupersededUpdates
        }

    }
}

Decline-Updates -WSUSServer $WSUSServer -WSUSPort $WSUSPort -reportonly:$reportonly
```

Uruchomienie z `-reportonly` pokazuje, co skrypt by odrzucił, bez wykonywania żadnej zmiany — wygodne przy pierwszym uruchomieniu na nieznanym serwerze, zanim zaufamy mu na produkcji.

WSUS = Windows Server Update Services: firmowy serwer aktualizacji Windows · **ARM64** = 64-bitowa architektura procesorów ARM (aarch64): inna niż x86-64 w zwykłych serwerach i komputerach; serwery P4 i Lore mają dla niej osobne wydania · **SP** = Special Publication: oznaczenie serii dokumentów NIST, tu wytycznych o hasłach SP 800-63B

TYLKO ODCZYT

Uruchomienie czyszczenia WSUS ręcznie: Invoke-WsusServerCleanup

```
Invoke-WsusServerCleanup -CleanupObsoleteComputers -CleanupUnneededContentFiles -
CleanupObsoleteUpdates -CompressUpdates -DeclineExpiredUpdates -DeclineSupersededUpdates
```

Skrypt wyżej wywołuje czyszczenie WSUS automatycznie, ale te same parametry można uruchomić osobno, ręcznie.

WSUS = Windows Server Update Services: firmowy serwer aktualizacji Windows

◫ TYLKO ODCZYT

Reindeksacja bazy SUSDB: sqlcmd

```
sqlcmd -S NazwaSerwera\NazwaInstancji -d SUSDB -i .\WsusDBMaintenance.sql
```

> (gdzie `NazwaSerwera\NazwaInstancji` to instancja SQL Server albo Windows Internal Database trzymająca `SUSDB`, a `WsusDBMaintenance.sql` to skrypt reindeksujący udostępniany przez Microsoft dla WSUS)

Po dużym czyszczeniu warto zreindeksować tabele w `SUSDB` — usunięcie tysięcy wierszy zostawia mocno pofragmentowane indeksy, które spowalniają samą konsolę WSUS. Uruchamiamy standardowy skrypt reindeksacji Microsoftu przeciwko bazie `SUSDB` (na WID albo na pełnym SQL Server, w zależności od instalacji).

◫ **SQL** = Structured Query Language: język zapytań do relacyjnych baz danych · **WSUS** = Windows Server Update Services: firmowy serwer aktualizacji Windows · **sqlcmd** = SQL command: klient bazy MSSQL

◫ ZMIENIA SYSTEM

Utworzenie zadania cyklicznego: New-ScheduledTaskAction + Register-ScheduledTask

```
$action = New-ScheduledTaskAction -Execute "powershell.exe" -Argument "-File
C:\Skrypty\Decline-Updates.ps1 -WSUSServer WSUSServer.Company.com -WSUSPort 8530"
$trigger = New-ScheduledTaskTrigger -Weekly -DaysOfWeek Sunday -At 2am
Register-ScheduledTask -TaskName "WSUS - odrzucanie i czyszczenie" -Action $action -
Trigger $trigger -RunLevel Highest -User "NazwaDomeny\NazwaKonta"
```

Zarówno odrzucanie starych paczek, jak i reindeksację warto uruchamiać automatycznie zamiast pamiętać o nich ręcznie.

◫ **WSUS** = Windows Server Update Services: firmowy serwer aktualizacji Windows

Z życia wzięte

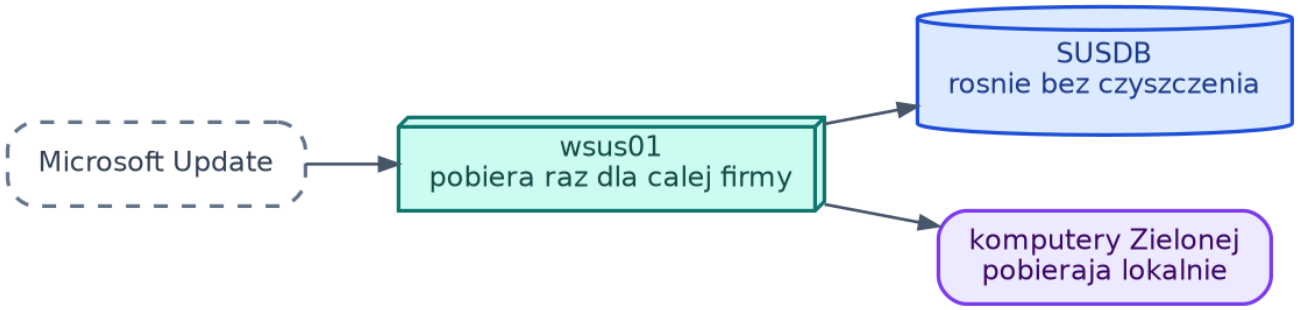
Z ŻYCIA WZIĘTE

We wtorek po Patch Tuesday wsus01 rozdaje najnowsze poprawki na wszystkie stacje w Pamiątkowie i Swarzędzu naraz, a konsola WSUS reaguje z zauważalnym opóźnieniem przy każdym kliknięciu. Sprawdzamy `SUSDB` i baza od dwóch lat nie była czyszczona, bo nikt nie ustawił tego jako zadanie cykliczne. Po uruchomieniu skryptu odrzucającego zastąpione aktualizacje i `Invoke-WsusServerCleanup` konsola znów odpowiada normalnie.

Czym jest WSUS i dlaczego jego baza puchnie

Pojęcie: WSUS. Windows Server Update Services — rola serwerowa, która pobiera aktualizacje z Microsoft Update raz, do jednego miejsca w sieci lokalnej, i rozprowadza je do klientów przez wewnętrzny serwer, zamiast każda maszyna pobierała te same pliki niezależnie z internetu. Metadane aktualizacji (tytuły, zależności, informacje o zastępowaniu) trzyma w bazie `SUSDB`, a same pliki instalacyjne — w katalogu na dysku.

Baza puchnie, bo WSUS domyślnie nie usuwa niczego sam z siebie. Każda aktualizacja, która została zastąpiona nowszą wersją (superseded), każdy wygasły biuletyn, każdy język, którego nie używamy — wszystko to zostaje w bazie i na dysku, dopóki ktoś ręcznie nie każe WSUS-owi to posprzątać. Po kilku latach eksploatacji bez czyszczenia baza `SUSDB` potrafi urosnąć do rozmiaru, przy którym konsola WSUS zaczyna reagować z zauważalnym opóźnieniem.



Aktualizacje ida od Microsoftu przez wsus01 do komputerow Zielonej, a SUSDB rosnie bez czyszczenia (kliknięcie otwiera pełny rozmiar)

Odrzucenie zastąpionych aktualizacji

Ręczne przeglądanie tysięcy aktualizacji w konsoli i odrzucanie ich pojedynczo nie ma sensu — poniższy skrypt robi to automatycznie, według zestawu reguł.

Wymagania: uprawnienia administracyjne do serwera WSUS, zainstalowany moduł PowerShell dla WSUS, połączenie z serwerem WSUS.

Zakres działania. Skrypt odrzuca aktualizacje, jeśli spełniają którykolwiek z poniższych warunków: są zastąpione przez nowsze aktualizacje (superseded); są oznaczone jako wygasłe (expired) według definicji Microsoft; dotyczą systemów operacyjnych x86 lub Itanium; są przeznaczone dla Windows XP; są pakietami językowymi (poza polskim i angielskim); dotyczą starszych wersji Internet Explorer (7, 8, 9); są aktualizacjami specyficznymi dla regionów; są aktualizacjami beta; są przeznaczone dla systemów wbudowanych (embedded).

Jeżeli jedna aktualizacja obejmuje wiele systemów operacyjnych i tylko część z nich spełnia powyższe kryteria, odrzucone zostają wyłącznie te wersje, które kryteria spełniają.

Invoke-WsusServerCleanup — parametry

Parametr	Co robi
`-CleanupObsoleteComputers`	usuwa z bazy komputery, które nie łączyły się z WSUS od dawna (odinstalowane, wymienione)
`-CleanupUnneededContentFiles`	usuwa z dysku pliki aktualizacji, których żadna zatwierdzona aktualizacja już nie potrzebuje
`-CleanupObsoleteUpdates`	usuwa z bazy metadane aktualizacji odrzuconych i wygasłych
`-CompressUpdates`	kompresuje zrewidowane aktualizacje, zmniejszając rozmiar bazy
`-DeclineExpiredUpdates`	odrzuca aktualizacje oznaczone przez Microsoft jako wygasłe
`-DeclineSupersededUpdates`	odrzuca aktualizacje zastąpione nowszymi

Częste błędy

Uruchamianie czyszczenia w środku dnia roboczego. `Invoke-WsusServerCleanup` na dużej, dawno nieczyszczonej bazie potrafi trwać wiele godzin i mocno obciąża serwer WSUS — konsola w tym czasie reaguje wolno albo w ogóle się nie otwiera. Zadanie planujemy na noc albo weekend.

Czyszczenie bez wcześniejszego odrzucenia starych aktualizacji. `Invoke-WsusServerCleanup` bez `-DeclineSupersededUpdates` / `-DeclineExpiredUpdates` posprząta tylko to, co już odrzucone — bez skryptu odrzucającego większość zbędnych aktualizacji nigdy nie trafi do sprzątania.

Powiązane: Windows Update z wiersza poleceń · WinSxS: czyszczenie magazynu komponentów

Źródła: notatki własne, skrypt bazowany na przykładzie Nicka Ealesa (Microsoft), dokumentacja `Invoke-WsusServerCleanup`

Revision #39
Created 2025-03-10 16:05:13 UTC by Tomasz Konieczny
Updated 2026-10-18 23:01:35 UTC by Tomasz Konieczny